Data Structures - Assignment no. 7

Date due: 10/9/02

Remarks:

- For every algorithmic problem, try to find the most efficient solution. Analyze the complexity of every algorithm you give, and in case it is not trivial, prove its correctness.
- Unless you are explicitly asked otherwise, please describe algorithms only to the extent which makes them clear to a reasonable reader. You may use plain language, or pseudo-code, whichever is more convenient.

1. Let $h : U \to \{1, \ldots, m\}$ be a function, and suppose that
$$\forall\, i \quad \Pr\left[h(x) = i\right] = \frac{1}{m}$$
Prove that h satisfies the simple uniform hashing assumption.

2. Consider the algorithm taught in class for finding the k'th order statistic in a given list of numbers $a_1, \ldots, a_n$, and analyze its expected running time, in the case where the pivot of each partition is selected randomly.

3. Given a sequence $a_1, \ldots, a_n$ of numbers, where the number of distinct values is guaranteed to be at most $\frac{n}{\log n}$, show a randomized algorithm that sorts the sequence in linear expected time.

4. Let Z be a random variable, whose values are non-negative integers. The expectation of Z is defined by
$$\mathbb{E}[Z] \stackrel{def}{=} \sum_{k=0}^{\infty} k \Pr\left[Z = k\right]$$
Suppose that the expectation of Z is finite, and prove that for every integer k,
$$\Pr[Z > k] \leq \frac{\mathbb{E}[Z]}{k+1}$$

5. Let H be a universal collection of functions from U to $\{1, \ldots, m\}$, and suppose that $|U| > m^2$. Prove that for every $i \in \{1, \ldots, m\}$ there exists a function $h \in H$ and an element $x \in U$, such that $h(x) = i$.

6. Given pointers to two nodes x, y in a tree, show an algorithm which finds the closest common ancestor for x and y. Your algorithm should run in time $O(d)$, where d is the ditance between x and y in the tree.